

WHAT IS CLAIMED IS:

1. A method for one pass assembly in raster image processing of elements using memory comprising the steps of:

5 forming a plurality of lists from elements within a job file including at least a first list for recurring elements and storing the first list in rasterized form, and at least a second list for variable elements and storing the second list in non-rasterized form;

identifying placement within at least one memory area of recurring and variable elements;

10 initializing the least one memory area with stored elements from the first list; and

raster image processing (RIPping) the memory area with elements from the second list.

15 2. The method of claim 1, further comprising the step of placing recurring elements in the memory areas after the step of raster image processing.

3. The method of claim 2, wherein the step of placing further comprises placing recurring elements that have no variable elements below them.

20 4. The method of claim 1, wherein the step of forming further comprises forming the first list with recurring elements that do not have any variable elements below them.

25 5. The method of claim 1, wherein the step of forming further comprises forming the second list from variable elements that are not recurring and from recurring elements that have variable elements below them.

30 6. The method of claim 5, wherein the step of forming further comprises forming the second list from variable elements that have variable elements both below and above them.

7. The method of claim 1, wherein the step of identifying further comprises locating overlapping areas between elements.

5 8. The method of claim 7, wherein the step of identifying further comprises identifying clip regions for calculating overlapping areas between elements.

9. The method of claim 8, wherein the step of identifying further comprises identifying clip regions that are non-rectangular to calculate overlapping areas.

10 10. The method of claim 7, wherein the step of identifying further comprises employing information from the job file to locate overlapping areas between elements.

15 11. The method of claim 10, wherein the step of RIPping further comprises RIPping elements from the second list into the memory in accordance with overlapping areas designated by identifying step and element placement within the job file.

20 12. The method of claim 1, further comprising the step of interpreting mark up language and page description language with the job file.

13. A raster image processing system for creating personalized prints comprising:

25 a print engine within the system that receives digital data to create prints; an input area within the system for receiving a pre-authored job file; processing means with associated storage means within the system coupled both to the print engine and the input area, for forming and storing a plurality of lists from elements within the pre-authored job file, including a list of recurring elements and a list
30 of variable elements, the list of recurring elements being stored in rasterized form and the list of variable elements in non-rasterized form;

at least one memory area within the system and associated with the processing means, the memory area being initialized with elements from the list of recurring elements and raster image processed using elements from the list of variable elements in accordance with element placement within the pre-authored job file;

5 a memory output device that allows the rasterized memory to be sent to the memory area.

14. The raster image processing system of claim 13, wherein the lists further include a second list of recurring elements that have variable elements below them, the second list of recurring elements being stored by the storage means in rasterized form and placed in the memory area on top of the variable elements in accordance with element placement within the pre-authored job file.

15. The system of claim 13, wherein the processing means reads the pre-authored job file including a mark up language and page description language.

16. The system of claim 13, wherein the list of recurring elements formed by the processing means further comprises elements that do not have any variable elements below them.

20 17. The system of claim 13, wherein the list of variable elements further comprises non-recurring elements and recurring elements that have variable elements above and below them.

25 18. The system of claim 13, wherein the memory area further comprises a plurality of memory bands.

19. The system of claim 18, wherein one of the memory bands is being initialized and RIPped with data from the pre-authored job file while the another memory 30 band is having its contents sent to the print engine by the memory output device.

100-6391-102301

20. The system of claim 13, wherein the processing means identifies overlapping areas between elements contained within the pre-authored job file.

21. The system of claim 13, wherein the memory area further comprises a plurality of memory tiles.

22. A method for raster assembly comprising the steps of:
5 forming a plurality of lists from elements within a job file including at least a first list for recurring elements and, at least a second list for variable elements;
10 identifying placement of the reoccurring elements and the variable elements as indicated in the job file for placement in at least one memory area;
initializing the memory area with the reoccurring elements and the variable elements in accordance with results from the identifying step; and
raster image processing (RIPping) the memory area.

15

23. The method of claim 22, wherein the step of RIPping further comprises prerasterizing all the elements allowing the memory areas to be used as a raster assembly tool.

20

24. The method of claim 22, wherein the step of RIPping allows one pass assembly and RIP processing of rasterized elements and PDL elements using banded memory.

25

25. The method of claim 23, wherein the step of identifying placement further comprises identifying overlapping elements.

26. The method of claim 25, wherein the step of identifying placement further comprises placing elements that contain transparent pixels.

30

27. The method of claim 22, wherein the step of forming further comprises adding image masks to recurring element.

28. The method of claim 22, wherein the step of RIPping further comprises RIPping elements on distributed computers.

5 29. The method of claim 22, further comprising following the step of forming the step of storing the rasterized version of recurring elements in either a lossy or losslessly compressed mode.

10 30. The method of claim 22, further comprising the step of storing elements in a raster-equivalent graphics state that allows the elements to be reused and rotated.

15 31. The method of claim 22, wherein the step of forming includes forming the first list with recurring elements having no variable elements below them and forming the second list with variable elements that are not prerasterized and further comprising the steps of:

forming a third list having recurring elements that have variable elements above and below them;

20 presetting the memory area with elements from the second list; and RIPping the third list and placing the RIPped third list elements into the memory area.

25 32. The method of claim 22, wherein the step of forming further comprises in addition to the first list and the second list, a third list containing elements that either are not prerasterized or elements that are rasterized and must be RIPped again due to layering consideration as well as a fourth list that are recurring but have variable elements beneath them.

30 33. The method of claim 32, wherein following the step of forming is a step of prerasterizing recurring elements from the first list.

40004603-202207

34. The method of claim 33, wherein the step of initializing further comprises presetting the memory areas with elements from the second list.

35. The method of claim 34, wherein the step of RIPping further
5 comprises RIPping the elements from the third list.

36. The method of claim 35, further comprising the step of applying the elements from the fourth list to the memory area.